An Efficient and Secure Data Sharing Scheme for Mobile Devices in Cloud Storage

D. Bujji Babu[1], Paleti Jagadeesh[2]

#1 Professor in The Department of MCA at QIS College of Engineering and Technology. (Autonomous), Vengamukkapalem, Prakasam (DT)

#2 PG Scholar in The Department of MCA QIS College of Engineering and Technology (Autonomous), Vengamukkapalem, Prakasam (DT).

ABSTRACT:

Clients can get a handy cloud storage solution thanks to mobile cloud storage (MCS). In this article, we provide a productive, safe, and privacy-preserving mobile cloud storage system that concurrently safeguards data confidentiality and privacy, particularly the access pattern. In particular, we provide an OSU protocol as the fundamental building block of the suggested mobile cloud storage system. By creating a small encrypted vector, OSU, which is based on onion additively homomorphic encryption with constant encryption layers, enables the client to covertly obtain an encrypted data item from the cloud and update it with a new value. It considerably lowers both the compute and communication overheads for the client. Our work is superior than earlier research in that it has useful characteristics, such as a fine-grained data structure (small item size), low client-side computation (a few additively homomorphic operations), and constant communication overhead, which make it more appropriate for MCS scenarios. Our system can also be verified to withstand malicious cloud by using the "verification chunks" method. The comparison and evaluation show that, in terms of client and cloud workloads, respectively, our plan is more effective than currently available oblivious storage options.

1.INTRODUCTION

Data in mobile cloud storage (MCS) is stored in the cloud and accessible via mobile devices from any location. Because of the appealing properties, MCS is turning out to be increasingly well known.

A few huge organizations give MCS administrations to business inspirations, for example Apple I Cloud, Drop box, Microsoft One Drive and Google Drive. As a rule, the cloud isn't thought of as completely trusted. As a result, prior to uploading data to the cloud, the client may

employ encryption techniques to maintain data confidentiality. Nonetheless, in MSC-based applications, information forever be connected with specific data, for example, area data in area based administrations. In this present circumstance, which thing of information is being gotten to spills expansion data to the cloud server. The cloud may be able to deduce the client's operation and even the encrypted data's content by making use of this leaked access pattern data. A cloud can, for instance, identify approximately 80% of the search queries in a searchable encryption system by employing a general inference attack with access pattern leakage and minimal background knowledge [1]. Neglectful innovation, like negligent exchange (OT) [2], unaware capacity (operating system) [3] and unmindful irregular access machine (ORAM) [4], is a sort of innovation that can safeguard the two information and access design. In general, these technologies enable a client to access its outsourced data stored in an unreliable cloud without disclosing the items visited or the operations requested. Because of the great level protection conservation, these innovations have been broadly applied in different application situations, for example, accessible encryption [5]-[7], encoded secret volumes [8], [9], distributed storage [10]-[13], multi-party calculation [14]-[18], and so on. Notwithstanding, there are a few difficulties to utilize existing neglectful plans into MCS situation because of a few reasons. Right off the bat, cell phones are for the most part associated with the Web through remote organizations, for example, impromptu, LTE, and Wi-Fi. That implies the cell phones have restricted correspondence transfer speed to download and transfer information. Hence, a few plans endured by the notable correspondence transmission capacity above lower bound outcome O(log N) [4] can not be utilized into MCS because of the weighty correspondence above. 1 Besides, albeit present day cell phones, like cell phones and tablets, have altogether improvement as far as registering capacity, they actually can't contend with PCs or other strong gadgets. Convoluted calculation additionally lessens the battery duration of cell phones. Hence, a few plans in view of completely homomorphic encryption (FHE) [19] or multi-facet onion additively homomorphic encryption [20] are likewise not reasonable for MCS because of mind boggling client-side encryption and unscrambling calculation, despite the fact that they dodge the correspondence lower bound and accomplish consistent correspondence

data transfer capacity above. Thirdly, many existing absent schemesare likewise endured by the ale least powerful thing size. Least powerful thing size alludes to the negligible number of pieces in a successful thing of an unaware plan expected to meet the predefined correspondence intricacy (consistent or logarithmic). Ale thing size keeps the portable client from fine-grained getting to its own information. In addition, it likewise further builds the correspondence or calculation above of existing absent plans.

A few unaware plans consider to acquaint information territory with further develop productivity. Information territory uncovers the propensity of a client to get to its information throughout a brief time frame. Spatial territory and transient region are two commonplace sorts of reference region of information access. Spatial region alludes that the client might get to the close by information things assuming that a specific thing is gotten to. Fleeting territory alludes that the client will reuse information more than once inside a brief time frame. The amortized communication overhead when accessing a series of items is lower than when accessing one item independently in non-constant communication overhead oblivious schemes that take spatial locality into account [21]. Exploiting fleeting territory can likewise essentially further develop productivity of specific unaware plans since in the event that a thing is visited, it just requires lightweight calculation and correspondence to get to the thing again in a brief time frame. In any case, apparently, there is no connected work that has thought about fleeting area.

In this paper, we propose a proficient, secure and protection saving versatile distributed storage conspire. The following characteristics characterize the proposed plan: 1) safeguarding information secrecy and access design at the same time, 2) steady correspondence transfer speed above, 3) low client side calculation (a couple additively homomorphic encryption and unscrambling tasks), 4) little least powerful thing size (a few kilobytes for sensible information limit), 5) thinking about transient region, and 6) irrefutable (against vindictive cloud). In particular, we feature our commitments of this paper in the accompanying.

We characterize a two-party convention, for example unaware choice and update (OSU) convention, and present a substantial development of OSU convention. A client is able to update its encrypted data by obliviously retrieving it

from the cloud and adding a new value with OSU. Contrasted and different techniques, for example, PIR-Read consolidated PIR-Compose, OSU requires less correspondence and client calculation. For specific information size, the proposed OSU has O(1) correspondence intricacy and requires the client to execute least encryption and unscrambling activities. Additionally, the convention is of autonomous interest for other secure multi-party calculation application situations.

2.LITERATURE SURVEY

2.1 An efficient and secure data sharing scheme for mobile devices in cloud computing

Abstract

With the development of big data and cloud computing, more and more enterprises prefer to store their data in cloud and share the data among their authorized employees efficiently and securely. So far, many different data sharing schemes in different fields have been proposed. However, sharing sensitive data in cloud still faces some challenges such as achieving data privacy and lightweight operations at resource constrained mobile terminals. Furthermore, most data sharing schemes have no integrity verification mechanism, which would result in wrong computation results for users. To solve the problems, we propose an efficient and secure data sharing scheme for mobile devices in cloud computing. Firstly, the scheme guarantees security and authorized access of shared sensitive data. Secondly, the scheme realizes efficient integrity verification before users share the data to avoid incorrect computation. Finally, the scheme achieves lightweight operations of mobile terminals on both data owner and data requester sides.

2.2 Secure Storage and Data Sharing Scheme Using Private Blockchain-Based HDFS Data Storage for Cloud Computing

Abstract

The storage of a vast quantity of data in the cloud, which is then delivered via the internet, enables Cloud Computing to make doing business easier by providing smooth access to the data and eliminating device compatibility limits. Data that is in transit, on the other hand, may be intercepted by a man-in-the-middle attack, a known plain text assault, a selected cypher text attack, a related key attack, or a pollution attack. Uploading data to a single cloud might, as a result, increase the likelihood that the secret data would be damaged. A distributed file system extensively used in huge data analysis for frameworks such as Hadoop is known as the Hadoop Distributed File System, more

commonly referred to as HDFS. Because with HDFS, it is possible to manage enormous volumes of data while using standard hardware that is not very costly. On the other hand, HDFS has several security flaws that might be used for malicious purposes. This highlights how critical it is to implement stringent security measures to make it easier for users to share files inside Hadoop and to have a reliable system in place to validate the shared files' validity claims. The major focus of this article is to discuss our efforts to improve the security of HDFS by using an approach made possible by blockchain technology (hereafter referred to as BlockHDFS). To be more precise, the proposed BlockHDFS uses the Hyperledger Fabric platform, which was developed for business applications, to extract the most value possible from the data inside files to provide reliable data protection and traceability in HDFS. In the results section, the performance of AES is superior to that of other encryption algorithms because it ranges from 1.2 milliseconds to 1.9 milliseconds. In contrast, DES ranges from 1.3 milliseconds to 3.1 milliseconds, three milliseconds to 3.6 millimetres, RC2 milliseconds to 3.9 milliseconds, and RSA milliseconds to 1.4 milliseconds, with data sizes ranging from 910 kilos.

3. PROPOSED SYSTEM

An efficient, secure, and privacy-preserving mobile cloud storage strategy is presented in this paper. The following characteristics characterize the proposed plan: 1) safeguarding information privacy and access design at the same time, 2) steady correspondence transfer speed above, 3) low clientside calculation (a couple additively homomorphic encryption and unscrambling tasks), 4) little least powerful thing size (a few kilobytes for sensible information limit), 5) thinking about transient territory, and 6) unquestionable (against malignant cloud). In particular, we feature our commitments of this paper in the accompanying.

We characterize a two-party convention, for example unaware choice and update (OSU) convention, and present a substantial development of OSU convention. A client is able to update its encrypted data by obliviously retrieving it from the cloud and adding a new value with OSU. Contrasted and different techniques, for example, PIR-Read consolidated PIR-Compose, OSU requires less correspondence and client calculation. For specific information size, the proposed OSU has O(1) correspondence intricacy and requires the client to execute least

encryption and unscrambling activities. In addition, there are additional secure multi-party computation application scenarios for which the protocol is of independent interest

In light of the proposed OSU convention, we present a productive, secure and protection safeguarding portable distributed storage plot. The plan can at the same time safeguard information content and protect access design security. Contrasted and past works, our plan has little thing size, low client-side calculation, and steady correspondence above. We likewise bring worldly territory into our development to additional upgrade the effectiveness. Our scheme can be verifiable and resist malicious cloud by combining the "verification chunks" method. Besides, we assess our development and other related works and the exploratory exhibitions show that our plan is more proficient

3.1 IMPLEMENTATION

3.1.1 Data Owners

In this module, the data provider uploads their encrypted **Owners** data in the Cloud server. For the security purpose the user encrypts the data file and then store in the server. The User can have capable of manipulating the encrypted data file and performs the following operations Register and Login, Upload Blocks, Verify Block (Data Auditing), Update Block, Delete File, View Uploaded Blocks.

3.1.2 Cloud Server

The **Cloud** server manages which is to provide data storage service for the Data Owners.Data owners encrypt their data files and store them in the Server for sharing with data consumers and performs the following operations such as Login, View Data Owners, View End Users, View Hash Table, View File Request, View Transactions, View Attackers, View Results, View File Time Delay Results, View File Throughput Results.

3.1.3 End User

In this module, the user can only access the data file with the secret key. The user can search the file for a specified keyword and end user and can do the following operations like Register and Login, View All Data Owner Files, Request File, View File Response, Download File.

3.1.4 Auditor

In this module, the key issuer performs the following operations Login, View Hash Table, View Attackers, View File Updated or Deleted, View Results

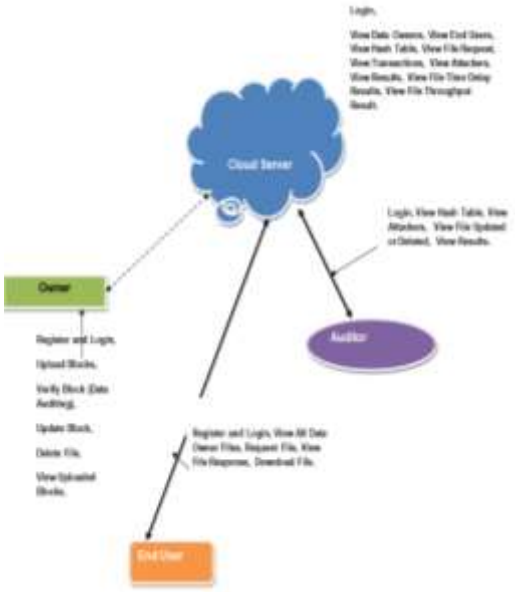

Fig 1: Architecture

4.RESULTS AND DISCUSSION

Performances of *Per Access* on 1 GiB Database

	Minimum effective item (block) size	Client comp.	Cloud comp.	Communication
Onion ORAM	1.20 MiB	25 min	> 3.5 days	7.4 MiB
C-ORAM	82.98 KiB	25.82 s	72 min	2.59 MiB
Ours	4.50 KiB	1.41 s	24.09 s	57.8 KiB

Table 1: performances

The Table 1 indicates the performances of per access in three schemes on a reasonable database (1 GiB). Due to the smaller item size and constant parameters, our scheme is much more efficient than Onion ORAM and C-ORAM in terms of client computation, cloud computation and communication. Since the runtime of Onion ORAM is enormous, the experimental results of Onion ORAM are estimated based on the evaluation performances of core operations in Onion ORAM. In our evaluation environment, even in the best case and omitting the amortized cost of eviction, the cloud computation of per access in Onion ORAM still takes about 3.5 days. Note that, if we repeatedly run access operation 18 times in our scheme to retrieve same size data (81 KiB) as in CORAM, our scheme has almost the same client computation overhead (25.38 s) as the C-ORAM scheme, but the cloud computation overhead is still much smaller than C-ORAM, i.e., about 10 times smaller.

5.CONCLUSION

In this paper, we propose a proficient, secure and protection safeguarding versatile distributed storage (MCS). The proposed plan can safeguard information and access design at the same time. Our scheme has a smaller item size, light client-side computation, and constant communication overhead compared to other schemes. To further boost the scheme's effectiveness, we also take temporal locality into account. Our

strategy has the potential to be demonstrated to resist malicious cloud by combining additional methods. We also present an oblivious selection and update protocol as part of the proposed MCS scheme. With this protocol, a client can use a small vector to obliviously select and update one of its encrypted data items outsourced in the cloud. Because of little client calculation and correspondence, we accept this convention might be of autonomous interest for other secure multi-party calculation application situations. The security and protection evidences and examinations show that our plan accomplishes information secrecy and adequate security conservation level. At long last, we contrast our plan and other two unaware stockpiling plans and completely gauge our development in a recreation climate. The findings demonstrate that our plan is highly effective and performs well.

REFERENCES:

[1] M. S. Islam, M. Kuzu, and M. Kantarcioglu, "Access pattern disclosure on searchable encryption: Ramification, attack and mitigation," in 19th Annual Network and Distributed System Security Symposium, NDSS 2012, San Diego, California, USA, February 5-8, 2012, 2012. [Online]. Available:

[2] J. Kilian, "Founding cryptography on oblivious transfer," in Proceedings of the 20th Annual ACM Symposium on Theory of Computing, May 2-4, 1988, Chicago, Illinois, USA, 1988, pp. 20–31. [Online]. Available:

[3] D. Boneh, D. Mazieres, and R. A. Popa, "Remote oblivious storage: Making oblivious ram practical," pp. 1–18, 2011.

[4] O. Goldreich and R. Ostrovsky, "Software protection and simulation on oblivious rams," J. ACM, vol. 43, no. 3, pp. 431–473, 1996. [Online]. Available

[5] J. Camenisch, M. Kohlweiss, A. Rial, and C. Sheedy, "Blind and anonymous identity-based encryption and authorised private searches on public key encrypted data," in Public Key Cryptography - PKC 2009, 12th International Conference on Practice and Theory in Public Key Cryptography, Irvine, CA, USA, March 18-20, 2009. Proceedings, 2009, pp. 196–214. [Online]. Available:

[6] T. Hoang, A. A. Yavuz, F. B. Durak, and J. Guajardo, "Oblivious dynamic searchable encryption via distributed PIR and ORAM," IACR Cryptology ePrint Archive, vol. 2017, p. 1158, 2017. [Online]. Available: [7] S. Garg, P. Mohassel, and C. Papamanthou, "TWORAM: efficient oblivious RAM in

two rounds with applications to searchable encryption," in Advances in Cryptology - CRYPTO 2016 - 36th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 14-18, 2016, Proceedings, Part III, 2016, pp. 563–592. [Online]. Available

[8] E. Blass, T. Mayberry, G. Noubir, and K. Onarlioglu, "Toward robust hidden volumes using write-only oblivious RAM, in Proceedings of the 2014 ACM SIGSAC Conference on Computer and Communications Security, Scottsdale, AZ, USA, November 3-7, 2014, 2014, pp. 203–214. [Online]. Available:

[9] D. S. Roche, A. J. Aviv, S. G. Choi, and T. Mayberry," Deterministic, stash-free write-only ORAM," in Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security, CCS 2017, Dallas, TX, USA, October 30 - November 03, 2017, 2017, pp. 507–521. [Online]. Available:

[10] E. Stefanov and E. Shi, "Oblivistore: High performance oblivious cloud storage," in 2013 IEEE Symposium on Security and Privacy, SP 2013, Berkeley, CA, USA, May 19-22, 2013, 2013, pp. 253–267. [Online]. Available:

[11] D. Cash, A. K¨upc¸ ¨ u, and D. Wichs, "Dynamic proofs of retrievability via oblivious RAM," in Advances in Cryptology EUROCRYPT 2013, 32nd Annual International Conference on the Theory and Applications of Cryptographic Techniques, Athens, Greece, May 26-30, 2013. Proceedings, 2013, pp. 279–295. [Online]. Available:

[12] E. Stefanov and E. Shi, "Multi-cloud oblivious storage," in 2013

ACM SIGSAC Conference on Computer and Communications Security, CCS'13, Berlin, Germany, November 4-8, 2013, 2013, pp. 247–258. [Online]. Available:

[13] B. Carbunar and R. Sion, "Write-once read-many oblivious RAM," IEEE Trans. Information Forensics and Security, vol. 6, no. 4, pp. 1394–1403, 2011.

[14] X. S.Wang, Y. Huang, T. H. Chan, A. Shelat, and E. Shi, "SCORAM: oblivious RAM for secure computation," in Proceedings of the 2014

ACM SIGSAC Conference on Computer and Communications Security, Scottsdale, AZ, USA, November 3-7, 2014, 2014, pp. 191–202. [Online]. Available:

[15] E. Boyle, K. Chung, and R. Pass, "Large-scale secure computation: multi-party computation for (parallel) RAM programs," in Advances in Cryptology - CRYPTO 2015 - 35th Annual Cryptology

Conference, Santa Barbara, CA, USA, August 16-20, 2015, Proceedings, Part II, 2015, pp. 742–762. [Online]. Available:

[16] C. Gentry, K. A. Goldman, S. Halevi, C. S. Jutla, M. Raykova, and D. Wichs, "Optimizing ORAM and using it efficiently for secure computation," in Privacy Enhancing Technologies - 13th International Symposium, PETS 2013, Bloomington, IN, USA, July 10-12, 2013. Proceedings, 2013, pp. 1–18. [Online]. Available:

[17] S. Lu and R. Ostrovsky, "Distributed oblivious RAM for secure two-party computation," in Theory of Cryptography - 10th Theory of Cryptography Conference, TCC 2013, Tokyo, Japan, March 3-6, 2013. Proceedings, 2013, pp. 377–396. [Online]. Available:

[18] S. Zahur, X. Wang, M. Raykova, A. Gasc´on, J. Doerner, D. Evans, and J. Katz, "Revisiting square-root ORAM: efficient random access in multi-party computation," in IEEE Symposium on Security and Privacy, SP 2016, San Jose, CA, USA, May 22-26, 2016, 2016, pp. 218–234. [Online]. Available [19] D. Apon, J. Katz, E. Shi, and A. Thiruvengadam, "Verifiable oblivious storage," in Public-Key Cryptography - PKC 2014 17th International Conference on Practice and Theory in Public-Key Cryptography, Buenos Aires, Argentina, March 26- 28, 2014. Proceedings, 2014, pp. 131–148. [Online]. Available:

[20] S. Devadas, M. van Dijk, C. W. Fletcher, L. Ren, E. Shi, and D. Wichs, "Onion ORAM: A constant bandwidth blowup oblivious RAM," in Theory of Cryptography 13th International Conference, TCC 2016-A, Tel Aviv, Israel, January 10-13, 2016, Proceedings, Part II, 2016, pp. 145–174. [Online]. Available:

AUTHOR PROFILE:

Dr.D.Bujji Babu, currently working as a Professor and Head in the Department of Master Of Computer Application, QIS College of Engineering and Technology, Ongole, Andhra Pradesh. He did his M. Tech (CSE) from JNTUK, Kakinada and Ph.D. (CSE) from Acharya Nagarjuna University. He published more than 50 research papers in reputed peer reviewed Scopus indexed journals. He also attended and presented research papers in different national and international journals and the proceedings were indexed IEEE, Springer Link series. He visited the countries Kuching, Malaysia for attending and presenting his research articles. 3 Patent

journals are published and in pipeline for grant. He wrote more than an dozen of monographs and published by the Technical Publishers. He Published Two Course content modules for the students of Acharya Nagarjuna University. He is the recognized research supervisor under JNTUK, Kakinada and guided several UG and PG Projects, currently supervising 3 research scholars under JNTUK. His area of interest is Software Engineering, Data Mining, Data Science, Big Data and Programming Languages. He is the Principal Investigator for the DST Sponsored Project and Co-PI for another DST Project.

Mr. P. Jagadeesh, PG Scholar in the department of MCA, QIS College of engineering and Technology

(Autonomous), Vengamukkapalem, Prakasam (DT) His areas of Interests are Networking & Cloud Computing.